Remarks

Reconsideration is requested in view of the preceding amendments and the following remarks.

By this Amendment, claims 1, 5, 7, 12, 23, 30, and 32 are amended and claims 15-22 are cancelled without prejudice. Upon entry of this Amendment, claims 1-5, 7-14, 23-30, and 32-38 are in the application.

Rejections in View of Yavatkar

Claims 1-4, 6-11, 13-17, 19-25, 27-36, and 38 stand rejected under 35 U.S.C. § 102(a) as allegedly anticipated by Yavatkar et al., "The Phoenix Framework: A Practical Architecture for Programmable Networks," IEEE Communications, pp. 160-165 (March 2000) (hereinafter "Yavatkar"). This rejection is traversed.

Yavatkar discloses a method of combating a network attack known as the TCP SYN attack in which an attacker floods a target machine with a continuous stream of TCP SYN packets. This packet stream causes the target machine to become busy, and can make the target machine unavailable. See Yavatkar at page 164, second column. Yavatkar's Phoenix framework allows a network administrator to trace the source of such an attack using a watchdog agent and a bloodhound agent. According to Yavatkar, if the watchdog agent is unable to establish a connection with a server, the watchdog agent raises an alert and launches a bloodhound agent. The bloodhound agent is configured with the address of the server being attacked and is sent to the active device topologically closest to the attacked server. The bloodhound agent then finds the attack ingress point.

Amended claim 1 recites a method for implementing intrusion detection that includes installing intrusion detection software on a plurality of remote computers and executing the intrusion detection software on the remote computers in response to a notice of a network intrusion. Yavatkar does not teach or suggest installing and executing intrusion detection software on a plurality of remote computers, but instead teaches installing intrusion tracking software (a bloodhound agent) only at an active device that is close to the location of the intrusion. Accordingly, claim 1 and dependent claims 2-5 and 7-14 are properly allowable.

Amended claim 23 recites a system for detecting intrusions in a computer network that includes an intrusion detection server configured to send a request to execute intrusion detection software to software agents at a plurality of computers when intrusion detection services are needed based on the at least one rule stored in said database. Yavatkar does not teach or suggest such a system. According to Yavatkar, a watchdog agent launches a bloodhound agent that is sent only to the active device topologically closest the intrusion. Yavatkar does not teach or suggest executing intrusion detection services on a plurality of computers. Accordingly, claim 23 and dependent claims 24-29 are properly allowable in view of Yavatkar.

Amended claim 30 recites, in part, receiving notification of a network intrusion and installing intrusion detection software on a plurality of remote computers in response to the received notification. Yavatkar teaches a bloodhound agent that is sent only to the active device topologically closest the intrusion, and claim 31 and dependent claims 32-38 are properly allowable over Yavatkar.

Rejections under 35 U.S.C. § 103 in View of Yavatkar and Porras

Claims 5, 12, 18, and 26-27 stand rejected as allegedly obvious from a combination of Yavatkar and Porras et al., U.S. Patent 6,484,203 ("Porras"). This rejection is traversed. The rejection of claim 18 is moot in view of the cancellation of claim 18 without prejudice. In addition, claims 5, 12, and 26-27 are properly allowable as dependent from allowable base claims. However, as described below, Yavatkar and Porras fail to teach or suggest the additional features of these dependent claims.

Amended claim 5 recites a method that includes, in part, monitoring for fulfillment of a stop condition, wherein the stop condition is an expiration time of a request to initiate intrusion detection services. The Office action states that Porras teaches such a stop condition at col. 9, lines 35-50. This portion of Porras is reproduced below:

> Intramonitor and intermonitor programming interfaces are substantially the same. These interfaces can be subdivided into five categories of interoperation: channel initialization and termination, channel synchronization, dynamic configuration, server probing, and report/event dissemination. Clients are responsible for initiating and terminating channel sessions with servers. Clients are also responsible for managing channel synchronization in the event of errors in message sequencing or periods of failed or slow response (i.e., "I'm alive" confirmations). Clients may also submit dynamic configuration requests to servers. For example, an analysis engine 22, 24 may request an event collection method to modify its filtering semantics. Clients may also probe servers for report summaries or additional event information. Lastly, servers may send clients intrusion/suspicion reports in response to client probes or in an asynchronous dissemination mode.

This cited portion of Porras does not teach or suggest a stop condition that is an expiration time. According to the Office action, terminating event monitoring and analysis because of slow response "meets the recitation of wherein the stop condition is an expiration time." This is incorrect. Termination based on slow response is not termination based on an expiration time.

The Office action states that Porras at column 5, lines 35-45 teaches continuous measures for intrusion detection. However, continuous measures for intrusion detection are not the same as stopping intrusion detection as claimed. Because Yavatkar and Porras fail to teach stop condition that is an expiration time for intrusion detection services, claim 5 is properly allowable.

Claim 12 is directed to a method for implementing an intrusion detection system that includes receiving a request to initiate intrusion detection services, wherein the request includes an expiration time indicating when to stop executing the intrusion detection software. As noted above, Porras does not teach or suggest such an expiration time, and claim 12 is properly allowable.

Claim 37 recites an article of manufacture comprising a computer-readable medium having stored thereon instructions adapted to be executed by a processor. The instructions which, when executed, define a series of steps including transmitting an installation request and installing intrusion detection software on a plurality of remote computers. The request includes a stop condition that is an expiration time indicating when to stop executing the intrusion detection software. As noted above, Porras fails to teach or suggest such an expiration time, and claim 37 is properly allowable.

Conclusion

In view of the preceding amendments and remarks, all pending claims are in condition for allowance and action to such end is requested.

Respectfully submitted,

KLARQUIST SPARKMAN, LLP

By _____
Michael D. Jones
Registration No. 41,879

One World Trade Center, Suite 1600
121 S.W. Salmon Street
Portland, Oregon 97204
Telephone: (503) 595-5300
Facsimile: (503) 228-9446